22883

PATENT TRADEMARK OFFICE

This application is submitted in the name of the following inventor(s):

Inventor	Citizenship	Residence City and State
Mark MUHLESTEIN	United States	Tucson, Arizona

The assignee is Network Appliance, Inc., a California corporation having an office at 495 East Java Drive, Sunnyvale, CA 94089.

Title of the Invention

Decentralized Appliance Virus Scanning

Background of the Invention

1. Field of the Invention

This invention relates to virus scanning in a networked environment.

2. Related Art

Computer networking and the Internet in particular offer end users unprecedented access to information of all types on a global basis. Access to information

can be as simple as connecting some type of computing device using a standard phone line to a network. With the proliferation of wireless communication, users can now access computer networks from practically anywhere.

Connectivity of this magnitude has magnified the impact of computer viruses. Viruses such as "Melissa" and "I love you" had a devastating impact on computer systems worldwide. Costs for dealing with viruses are often measured in millions and tens of millions of dollars. Recently it was shown that hand-held computing devices are also susceptible to viruses.

Virus protection software can be very effective in dealing with viruses, and virus protection software is widely available for general computing devices such as personal computers. There are, however, problems unique to specialized computing devices, such as filers (devices dedicated to storage and retrieval of data). Off-the-shelf virus protection software will not run on a specialized computing device unless it is modified to do so, and it can be very expensive to rewrite software to work on another platform.

A first known method is to scan for viruses at the data source. When the data is being provided by a specialized computing device the specialized computing device must be scanned. Device-specific virus protection software must be written in order to scan the files on the device.

While this first known method is effective in scanning files for viruses, it suffers from several drawbacks. First, a company with a specialized computing device would have to dedicate considerable resources to creating virus protection software and maintaining up-to-date data files that protect against new viruses as they emerge.

Additionally, although a manufacturer of a specialized computing device could enlist the assistance of a company that creates mainstream virus protection software to write the custom application and become a licensee this would create other problems, such as reliance on the chosen vendor of the anti-virus software, compatibility issues when hardware upgrades are effected, and a large financial expense.

A second known method for protecting against computer viruses is to have the end user run anti-virus software on their client device. Anti-virus software packages are offered by such companies as McAfee and Symantec. These programs are loaded during the boot stage of a computer and work as a background job monitoring memory and files as they are opened and saved.

While this second known method is effective at intercepting and protecting the client device from infection, it suffers from several drawbacks. It places the burden of detection at the last possible link in the chain. If for any reason the virus is not detected prior to reaching the end user it is now at the computing device where it will do the most damage (corrupting files and spreading to other computer users and systems).

It is much better to sanitize a file at the source from where it may be delivered to millions of end users rather than deliver the file and hope that the end user is prepared to deal with the file in the event the file is infected. End users often have older versions of anti-virus software and/or have not updated the data files that ensure the software is able to protect against newly discovered viruses, thus making detection at the point of mass distribution even more critical.

Also, hand-held computing devices are susceptible to viruses, but they are poorly equipped to handle them. Generally, hand-held computing devices have very limited memory resources compared to desktop systems. Dedicating a portion of these resources to virus protection severely limits the ability of the hand-held device to perform effectively. Reliable virus scanning at the information source is the most efficient and effective method.

Protecting against viruses is a constant battle. New viruses are created everyday requiring virus protection software manufacturers to come up with new data files (solution algorithms used by anti-virus applications). By providing protection at the source of the file, viruses can be eliminated more efficiently and effectively.

Security of data in general is important. Equally important is the trust of the end user. This comes from the reputation that precedes a company, and companies that

engage in web commerce often live and die by their reputation. Just like an end user trusts that the credit card number they have just disclosed for a web-based sales transaction is secure they want files they receive to be just as secure.

Accordingly, it would be desirable to provide a technique for scanning specialized computing devices for viruses and other malicious or unwanted content that may need to be changed, deleted, or otherwise modified.

Summary of the Invention

The invention provides a method and system for scanning specialized computing devices (such as filers) for viruses. In a preferred embodiment, a filer is connected to one or more supplementary computing devices that scan requested files to ensure they are virus free prior to delivery to end users. When an end user requests a file from the filer the following steps occur: First, the filer determines whether the file requested must be scanned before delivery to the end user. Second, the filer opens a channel to one of the external computing devices and sends the filename. Third, the external computing device opens the file and scans it. Fourth, the external computing device notifies the filer the status of the file scan operation. Fifth, the filer sends the file to the end user provided the status indicates it may do so.

This system is very efficient and effective as a file needs only to be scanned one time for a virus unless the file has been modified or new data files that protect against new viruses have been added. Scan reports for files that have been scanned may be stored in one or more of the external computing devices, in one or more filers, and some portion of a scan report may be delivered to end users.

In alternative embodiments of the invention one or more of the external computing devices may be running other supplementary applications, such as file compression and encryption, independently or in some combination.

Brief Description of the Drawings

Figure 1 shows a block diagram of a system for decentralized appliance virus scanning.

Figure 2 shows a process flow diagram for a system for decentralized virus scanning

Detailed Description of the Preferred Embodiment

In the following description, a preferred embodiment of the invention is described with regard to preferred process steps and data structures. Those skilled in the art would recognize after perusal of this application that embodiments of the invention can be implemented using one or more general purpose processors or special purpose processors or other circuits adapted to particular process steps and data structures described herein, and that implementation of the process steps and data structures described herein would not require undue experimentation or further invention.

Lexicography

The following terms refer or relate to aspects of the invention as described below. The descriptions of general meanings of these terms are not intended to be limiting, only illustrative.

- Virus – in general, a manmade program or piece of code that is loaded onto a computer without the computer user's knowledge and runs against their wishes. Most viruses can also replicate themselves, and the more dangerous types of viruses are capable of transmitting themselves across networks and bypassing security systems.

- **client** and **server** — in general, these terms refer to a relationship between two devices, particularly to their relationship as client and server, not necessarily to any particular physical devices.

  For example, but without limitation, a particular client device in a first relationship with a first server device, can serve as a server device in a second relationship with a second client device. In a preferred embodiment, there are generally a relatively small number of server devices servicing a relatively larger number of client devices.

- **client device** and **server device** — in general, these terms refer to devices taking on the role of a client device or a server device in a client-server relationship (such as an HTTP web client and web server). There is no particular requirement that any client devices or server devices must be individual physical devices. They can each be a single device, a set of cooperating devices, a portion of a device, or some combination thereof.

For example, but without limitation, the client device and the server device in a client-server relation can actually be the same physical device, with a first set of software elements serving to perform client functions and a second set of software elements serving to perform server functions.

- **web client** and **web server** (or **web site**) — as used herein the terms "web client" and "web server" (or "web site") refer to any combination of devices or software taking on the role of a web client or a web server in a client-server environment in the internet, the world wide web, or an equivalent or extension thereof. There is no particular requirement that web clients must be individual devices. They can each be a single device, a set of cooperating devices, a portion of a device, or some combination thereof (such as for example a device providing web server services that acts as an agent of the user).

As noted above, these descriptions of general meanings of these terms are not intended to be limiting, only illustrative. Other and further applications of the invention, including extensions of these terms and concepts, would be clear to those of ordinary skill in the art after perusing this application. These other and further applications are part of the scope and spirit of the invention, and would be clear to those of ordinary skill in the art, without further invention or undue experimentation.

System Elements

Figure 1 shows a block diagram of a system for decentralized appliance virus scanning.

A system 100 includes a client device 110 associated with a user 111, a communications network 120, a filer 130, and a processing cluster 140.

The client device 110 includes a processor, a main memory, and software for executing instructions (not shown, but understood by one skilled in the art). Although the client device 110 and filer 140 are shown as separate devices there is no requirement that they be physically separate.

In a preferred embodiment, the communication network 120 includes the Internet. In alternative embodiments, the communication network 120 may include alter-

native forms of communication, such as an intranet, extranet, virtual private network, direct communication links, or some other combination or conjunction thereof.

A communications link 115 operates to couple the client device 110 to the communications network 120.

The filer 130 includes a processor, a main memory, software for executing instructions (not shown, but understood by one skilled in the art), and a mass storage 131. Although the client device 110 and filer 130 are shown as separate devices there is no requirement that they be separate devices. The filer 130 is connected to the communications network 120.

The mass storage 131 includes at least one file 133 that is capable of being requested by a client device 110.

The processing cluster 140 includes one or more cluster device 141 each including a processor, a main memory, software for executing instructions, and a mass storage (not shown but understood by one skilled in the art). Although the filer 130 and the processing cluster 140 are shown as separate devices there is no requirement that they be separate devices.

In a preferred embodiment the processing cluster 140 is a plurality of personal computers in an interconnected cluster capable of intercommunication and direct communication with the filer 130.

The cluster link 135 operates to connect the processing cluster 140 to the filer 130. The cluster link 135 may include non-uniform memory access (NUMA), or communication via an intranet, extranet, virtual private network, direct communication links, or some other combination or conjunction thereof.

Method of Operation

Figure 2 shows a process flow diagram for a system for decentralized appliance virus scanning.

A method 200 includes a set of flow points and a set of steps. The system 100 performs the method 200. Although the method 200 is described serially, the steps of the method 200 can be performed by separate elements in conjunction or in parallel, whether asynchronously, in a pipelined manner, or otherwise. There is no particular requirement that the method 200 be performed in the same order in which this description lists the steps, except where so indicated.

At a flow point 200, the system 100 is ready to begin performing the method 200.

At a step 201, a user 111 utilizes the client device 110 to initiate a request for a file 133. The request is transmitted to the filer 130 via the communications network 120. In a preferred embodiment the filer 130 is performing file retrieval and storage at the direction of a web server (not shown but understood by one skilled in the art).

At a step 203, the filer 130 receives the request for the file 133 and sends the file ID and path of the file 133 to the processing cluster 140 where it is received by one of the cluster device 141.

At a step 205, the cluster device 141 uses the file ID and path to open the file 133 in the mass storage 131 of the filer 130.

At a step 207, the cluster device 141 scans the file 133 for viruses. In a preferred embodiment, files are tasked to the processing cluster 140 in a round robin fashion. In alternative embodiments files may be processed individually by a cluster device 141, by multiple cluster device 141 simultaneously, or some combination thereof. Load balancing may be used to ensure maximum efficiency of processing within the processing cluster 140.

There are several vendors offering virus protection software for personal computers, thus the operator of the filer 130 may choose whatever product they would like to use. They may even use combinations of vendors' products in the processing cluster 140. In an alternative embodiment of the invention, continual scanning of every file 133 on the filer 130 may take place.

The processing cluster 140 is highly scaleable. The price of personal computers is low compared to dedicated devices, such as filers, therefore this configuration is very desirable. Additionally, a cluster configuration offers redundant systems availability in case a cluster device 141 fails – failover and takeover is also possible within the processing cluster.

At a step 209, the cluster device 141 transmits a scan report to the filer 130. The scan report primarily reports whether the file is safe to send. Further information may be saved for statistical purposes (for example, how many files have been identified as infected, was the virus software able to sanitize the file or was the file deleted) to a database. The database may be consulted to determine whether the file 133 needs to be scanned before delivery upon receipt of a subsequent request. If the file 133 has not changed since it was last scanned and no additional virus data files have been added to the processing cluster, the file 133 probably does not need to be scanned. This means the file 133 can be delivered more quickly.

Other intermediary applications may also run separately, in conjunction with other applications, or in some combination thereof within the processing cluster 140. Compression and encryption utilities are some examples of these applications. These types of applications, including virus scanning, can be very CPU intensive, thus outsourcing can yield better performance by allowing a dedicated device like a filer to do what it does best and farm out other tasks to the processing cluster 140.

At a step 211, the filer 130 transmits or does not transmit the file 133 to the client 110 based on its availability as reported following the scan by the processing cluster 140. Some portion of the scan report may also be transmitted to the user.

At this step, a request for a file 133 has been received, the request has been processed, and if possible a file 133 has been delivered. The process may be repeated at step 201 for subsequent requests.

Generality of the Invention

The invention has wide applicability and generality to other aspects of processing requests for files.

The invention is applicable to one or more of, or some combination of, circumstances such as those involving:

- file compression;
- file encryption; and
- general outsourcing of CPU intensive tasks from dedicated appliances to general purpose computers.

Alternative Embodiments

Although preferred embodiments are disclosed herein, many variations are possible which remain within the concept, scope, and spirit of the invention, and these variations would become clear to those skilled in the art after perusal of this application.

Docket No. 103.1056.01